PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F

A2

(11) International Publication Number: **WO 98/40810**

(43) International Publication Date: 17 September 1998 (17.09.98)

(21) International Application Number: PCT/US98/04910

(22) International Filing Date: 12 March 1998 (12.03.98)

(30) Priority Data:
08/815,739 12 March 1997 (12.03.97) US

(71) Applicant: STORAGE TECHNOLOGY CORPORATION [US/US]; 2270 South 88th Street, Louisville, CO 80028 (US).

(72) Inventors: NGUYEN, Thai; 2638 E. 102nd Avenue, Thornton, CO 80229 (US). RAYMOND, Robert, Michael; 6133 Songbird Circle, Boulder, CO 80303 (US).

(74) Agents: SCHWARTZ, Paul, M. et al.; Brooks & Kushman, 22nd floor, 1000 Town Center, Southfield, MI 48075 (US).

(81) Designated States: JP, European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
Without international search report and to be republished upon receipt of that report.

(54) Title: NETWORK ATTACHED VIRTUAL TAPE DATA STORAGE SUBSYSTEM

(57) Abstract

The network attached virtual tape storage subsystem interconnects a plurality of tape devices with a plurality of data processors via a high bandwidth switching network to implement a virtual, distributed tape data storage subsystem. The virtual, distributed tape data storage system realizes multiple virtual devices, which are available to any of the data processors and the bandwidth of the system is scalable and can be changed on demand. The virtual tape storage subsystem is managed by a system controller which contains a plurality of software elements including: resource allocation, resource configuration, and resource management. The use of a networked storage manager enables the tape devices to be managed as a pool and yet attach the tape devices directly to the network as individual resources. The networked storage manager must provide the mechanism for the enterprise management to control tape device allocation and configuration as well as other functions, such a tape cartridge movement and data migration.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

NETWORK ATTACHED VIRTUAL TAPE DATA STORAGE SUBSYSTEM

Field of the Invention

This invention relates to tape data storage systems and, in particular, to a plurality of tape devices which are connected to a plurality of data processors via a network and which collectively implement a virtual, distributed tape data storage subsystem. The virtual, distributed tape data storage subsystem realizes multiple virtual devices, which are available to any of the data processors via the network which has a scalable bandwidth and can be changed on demand.

Problem

It is a problem in the field of data storage subsystems to provide adequate data storage service to the data processors that are connected to the data storage devices. There are numerous data storage media in use as well as corresponding data storage subsystem configurations which attempt to improve the data storage capabilities of the data storage media that is used to implement the data storage devices. For example, increases in the areal density of data storage products translate into lower data storage costs per bit, but do not always yield higher data transfer rates. To achieve increased data transfer rates requires architectural approaches to data storage rather than data storage device improvements. One other aspect of this data storage problem is that the allocation of customer data to a single type of data storage media represents a limitation when faced with widely varying data storage needs. This limitation can be partly obviated by balancing I/O activity across an array of data storage devices of a data storage subsystem. However, a fixed array configuration of data storage devices also limits the scalability in performance and provides no facility for applications to request changes in performance. An architecture where the data storage devices are located behind a server further limits the delivered performance since the bandwidth is limited by the server itself. Therefore, architecting a data storage subsystem that can efficiently serve the needs of the applications extant on the data processors is a daunting problem. There are numerous factors which effect performance and this problem is particularly pertinent to tape devices, since the tape media is experiencing significant enhancements to its data storage capacity.

The traditional tape device is directly connected to a single data processor in a dedicated tape device configuration. The data processor has exclusive use of

the tape device and typically communicates with the tape device via a SCSI interface. However, the use of dedicated tape devices is an expensive proposition where there are a plurality of data processors to be served, especially if the data access loads generated by the plurality of data processors are erratic. In this data storage subsystem architecture, the utilization of the tape devices and the efficiency of the data storage function are less than optimal, since each data processor is limited to its dedicated tape device and its physical constraints.

An alternative data storage subsystem architecture is to connect a plurality of tape devices along with a plurality of data processors to a common data communication network. In this architecture, the data processors all have access to all of the tape devices. The data processors run tape server software to manage the access protocol for the plurality of tape devices. Among the problems with the network interconnected tape devices is that it is difficult to share a tape device among a plurality of data processors. To provide enhanced response time, the tape devices can be served by an automated tape cartridge library system which mounts/dismounts the tape cartridges for the plurality of tape devices served by the automated tape cartridge library system. However, the tape cartridge library systems typically have a SCSI interface in the data path, and the SCSI interface introduces a number of physical limitations to the operation of the automated tape cartridge library system. The first limitation is that only a small number of tape devices can be attached to a SCSI bus compared to other bus architectures. The second limitation is the limited bandwidth of the SCSI bus that is shared by these tape devices. The length of the SCSI cable may also represent an additional limitation, since the length of the SCSI bus is typically limited to 25 feet.

A variation of this network data storage architecture is the use of a plurality of tape devices configured into a tape array. The tape devices are configured in a redundant array of data storage devices in a manner analogous to the Redundant Array of Inexpensive Disks (RAID) which is a well known architecture in the disk device technology. The tape array is typically located behind a server, which is directly connected to the network, and the bandwidth for data transfers between the data processors and the tape array is not scalable and is also limited by the

characteristics of the server. The tape array itself is also not scalable or easily changed due to the server limitation.

These various tape device based data storage subsystem architectures are all limited in their data exchange bandwidth and are restricted by the use of a single data storage media. The alternative to a single media data storage subsystem is the integration of a plurality of data storage media types and data storage subsystem architectures into a single data storage subsystem, typically termed a mass storage subsystem. One such data storage subsystem designed to address this problem is the mass storage system described in the paper by Sam Coleman and Steve Miller titled "Mass Storage System Reference Model: Version 4", published May 1990 by the IEEE Technical Committee on Mass Storage Systems and Technology. This mass storage system interconnects a plurality of data processors with a diversity of data storage subsystems via a high bandwidth switched network for the transmission of data therebetween at high data transfer rates. A separate network is used to interconnect the data processors with the mass storage system controller, which manages the processing of data transfer requests received from the data processors over the control network. The mass storage system controller is directly connected to the controllers of the various data storage subsystems and transmits data file retrieval requests to the selected controller in response to the received data file request received from the data processors. The file staging process used therein copies a data file in its entirety from the mass storage system to the client data processor via the high bandwidth network before the user accesses any of the requested data. Usually the data file is staged from an archival storage device, but staging from a direct access storage device is possible as well. File staging is initiated by a client data processor transmitting a request to the mass storage system identifying the data file by name. The mass storage system maintains mapping information indicative of the physical memory storage location occupied by the requested data file. The retrieved mapping information is used by the mass storage system to transmit a file retrieval request to the archival storage device on which the requested data file is stored. Upon receipt of this request, the designated storage device copies the requested

data file over the high speed network to a local, direct access data storage device that serves the requesting client data processor.

A significant limitation of this architecture is that the data is managed on a data file basis. Each client data processor request for a data file causes the mass storage system to access the mapping tables to locate the requested data file. When a client data processor sequentially accesses a plurality of data files, the mass storage system must successively access the mapping tables to identify each requested data file. As the extent of the mass storage system data storage capacity increases, the size and extent of the mapping tables proportionately increases and the time required to retrieve mapping information for each data file becomes a significant performance limitation of the mass storage system.

An improvement to the mass storage system is disclosed in the paper by J.L. Sloan et al, titled "MaSSIVE™: The Mass Storage System IV Enterprise", published April 1993 in the Proceedings of the IEEE, Vol. 81, No 4 and also disclosed in U.S. Patent No. 5,566,331. The MaSSIVE mass storage system stages entire file-systems as bit files between archival storage devices and direct access storage devices. Because these direct access storage devices are channel-attached to the client data processor, the file-systems contained thereon may be efficiently accessed by the client data processors exactly as if they were located on local data storage devices. Since entire file-systems are staged rather than individual data files, staging becomes less convenient to the user since multiple, associated file sets are staged together. On the other hand, by staging whole file-systems without interpretation to a storage device which is channel-attached to the client data processor, the inefficiencies and bottlenecks of network file service are avoided. Thus, this mass storage system design combines the benefits of file staging and network file service, but at the same time minimizes the drawbacks of each.

Thus, the use of a plurality of different types of media in a mass storage subsystem presents its own set of problems. With advances in massively parallel processing, there is a need for interconnection networks that provide high bandwidth inter-process communications and data storage configurations that match the high bandwidth network capacity. There is also a need for data storage devices that can scale in capability or capacity of data storage and data transfer

throughput. These data storage devices should be accessible via a high bandwidth switched network, such that a plurality of paths and links are provided between the data processor and the data storage devices. The data storage devices should be directly connectible to the switching network and not captive behind a server to thereby increase data throughput. The data transfers should be operational without the need to expend processing resources and the data storage devices should be shared among multiple data processors. The above-noted mass storage systems address the issue of utilizing the high bandwidth network that interconnects the data processors with the data storage subsystems to a high degree of efficiency. However, the data storage subsystems described therein represent archival data storage subsystems and are not efficiently used for simple routine data file access. In addition, the data storage image of the various data storage subsystems are immutable and the mass storage system controller simply functions as a data transfer manager to ensure that the data files or file systems are relocated from the archival data storage subsystem tot he local data storage maintained by the data processors. There is no attempt to address the issue of the local storage element used by the data processors. Thus, the local storage systems described in the mass storage system publications suffer the limitation described above with respect to the tape devices.

Solution

The above described problems are solved and a technical advance achieved in the field by the network attached virtual tape storage subsystem of the present invention. This invention relates to tape data storage systems and, in particular, to a plurality of tape devices which are connected to a plurality of data processors via a high bandwidth switching network and which collectively implement a virtual, distributed tape data storage subsystem. The virtual, distributed tape data storage system incorporates elements from the mass storage system technology as well as tape device data storage subsystem architectures to realize multiple virtual devices, which are available to any of the data processors and the bandwidth of the system is scalable and can be changed on demand. By pooling the tape devices together, and interconnecting them with the data processors via a fiber data network, the problems of prior art tape device data storage subsystem architectures are

overcome. This architecture realizes multiple virtual devices, which are available to any of the data processors. This architecture enables any tape array to be realized, including, but not limited to: RAIT 0, 1, 3, 4, 5, and the tape array and data transmission bandwidth can be dynamically reconfigured, since the network switchably interconnects the tape devices to the data processors.

The virtual tape storage subsystem is managed by a system controller which contains a plurality of software elements including: resource allocation, resource configuration, and resource management. The system controller may also contain security software for authentication. The resource allocation software has the responsibility to keep track of the resource usage such as which data processor presently owns the tape device, which tape devices are free to be allocated to the requesting data processors and the like. The resource configuration software allows an operator to configure the data storage resources for the data processors which are attached to the network. The operator can assign the maximum number of tape devices that a data processor can designate the configuration of these tape devices. The resource configuration software automatically configures both the tape devices allocated to a data processor as well as the connection between the data processor and a tape device(s). The resource management software queues the request for the resource allocation and notifies the data processor when the requested resource is ready, or it can schedule the availability of the resources.

The use of a networked storage manager enables the tape devices to be managed as a pool and yet attach the tape devices directly to the network as individual resources. The networked storage manager must provide the mechanism for the enterprise management to control tape device allocation and configuration as well as other functions, such as tape cartridge movement and data migration. The rules which could be implemented address response time constraints, data file transfer size, data file transfer rates, data file size bounds and the like. The networked storage manager manages the allocation, configuration and security.

Brief Description of the Drawing

Figure 1 illustrates in block diagram form the overall architecture of a traditional direct attached tape configuration;

Figure 2 illustrates in block diagram form the overall architecture of a traditional client/server network attached tape configuration;

Figure 3 illustrates in block diagram form the overall architecture of a traditional network attached RAIT tape configuration;

Figure 4 illustrates in block diagram form the overall architecture of a prior art network connected mass storage system which uses a multitude of data storage devices;

Figure 5 illustrates in block diagram form the overall architecture of an embodiment of the network attached virtual tape storage subsystem of the present invention; and

Figures 6 and 7 illustrate in block diagram form the overall architecture of the dynamic mapping and data striping functions, respectively, of the network attached virtual tape storage subsystem of the present invention.

Detailed Description

In order to avoid ambiguity in the description of the network attached, distributed, virtual tape storage subsystem, the following set of definitions are provided. These definitions represent industry accepted interpretations of the terms of art used in the data storage technology art.

Definitions

A channel is typically a specialized, high speed, point-to-point hardware connection between a data processor and a device controller. A channel allows the data processor to send requests represented in a particular controller's command language to that controller and to receive responses from the controller, in order to cause a storage device attached to that controller to access and alter data blocks stored on that storage device.

A controller is a hardware device onto which is attached one or more storage devices (either direct or sequential access). The storage devices attached to a particular controller can be uniquely identified by ordinal numbers. The controller maps requests by a data processor for operations on blocks into the hardware operations understood by the storage device. Requests to a controller from a data processor are represented in a command language understood by the controller.

A data file (or simply "file") is a sequential collection of bytes. Not all files

have the same number of bytes. Bytes may be added to or removed from a file, so that a single file may change in size. Each byte in a file may be uniquely identified by an ordinal number. The first byte in a file is byte zero. Files allow a collection of bytes to be operated on as a group.

The term data processor means the same thing as the terms computer, computing system, and computer system, and encompasses all forms of digital, stored program computers including but not limited to supercomputers, workstations, PCS, personal computers, minicomputers, mainframes, microcomputers, and servers. A device driver is a portion of the operating system which maps the file-system code I/O requests for operations on blocks inside a particular file-system into the appropriate I/O command language for the device controller to which is connected the direct access storage device which contains the particular file-system blocks being accessed. Different device controllers require different I/O command language, and an operating system frequently has several different device drivers, one for each different type of device controller. Only the device driver for a particular controller needs to understand the I/O command language for that controller. Because a device driver presents a common interface to the file-system code, regardless of the differences of the individual device controllers, the file-system code can issue a set of I/O requests that is common to all device controllers, even though the controllers may be different one from another.

A direct access storage device is a hardware device which stores a collection of blocks, and a mechanism which allows the accessing and/or alternating of any one of the blocks without accessing or altering any of the other blocks on the direct access storage device. Such a method of access means that the time or effort to access any particular block is approximately the same to access any other block on the device. This type of access is often referred to as random access. The number of blocks on a particular direct access storage device is fixed. All blocks on a particular direct access storage devices are the same size. The blocks on a direct access storage device are uniquely identified by an ordinal number, with the first block on the device identified by ordinal value zero. An example of a direct access storage device is a magnetic disk device.

A directory is a file that defines a set of files. By definition, a directory may also contain other directories, so that a hierarchical organization of files and directories may be defined. Each entry (file or directory) in a particular directory is uniquely identified by a file name. Files and directories may be added to and removed from a directory, or moved out of one directory and into another, so that directories may change in size, and not all directories are the same size. Directories allow a collection of files and directories to be operated on as a group. Since the information about what files or directories are contained in a directory is stored as bits within one or more blocks, just as the information in a file is stored as bits within one or more blocks, a directory can be seen as a special type of file. The difference is in how the bits that are part of the directory are interpreted.

File staging or simply staging is the process by which a data file, bit file or file-system is copied in its entirety from one storage device to another. Typically, staging occurs between archival storage devices and direct access storage devices. Staging can also occur between storage devices belonging to a client data processor and storage devices belonging to a mass storage system. The process of staging usually occurs at the beginning and ending of a file access operation. Typically, a file is staged from an archival device prior to a direct access storage device prior to data access by the client data processor, while the file is staged from a direct access storage device to an archival storage device after the client data processor is finished accessing the data.

A file-system is a self-contained, self-defining hierarchical collection of data files and directories, and is composed of a sequential collection of blocks which are uniquely identified by a set of ordinal numbers, with the first block identified by ordinal value zero. The structure of a file-system and the organization of the files contained within the file-system is defined solely by information contained in the file-system itself. The file-system contains all of the blocks composing all of the files and directories that are part of it. In addition, the file-system contains information defining which blocks belong to what files and directories, and in what order they occur inside each file and directory. This additional information is stored as bits in blocks on the storage device along with the blocks composing the files and directories inside the file-systems. Files and directories can be added to and

removed from a file-system, and they can be moved from one file-system to another. Thus, a file-system can be thought of as nothing more than a collection of blocks on a storage device, or if the contents of the file-system blocks are interpreted, a file-system may be thought of as self-contained, self-defining hierarchical collection of data files and directories. A particular file within the hierarchy of directories and files within the file-system may be identified within the hierarchy by naming the file's path name. A particular byte within any file can be identified by a combination of path name (to identify the specific file) and the ordinal number identifying the byte within the file.

Blocks are a physical organization imposed on collections of bits by hardware storage devices (both direct and sequential access) for reasons of ease of implementation. Bytes, files and directories are a conceptual organization imposed on collections of bits by software for the convenience of users. A file-system defines a mapping of the conceptual organization into the physical organization, by defining a mapping of files and directories onto blocks stored on a direct access storage device. A file-system allows a collection of files and directories to be stored, accessed, and altered, on storage device, and to be operated on as a group.

A mass storage system is a collection of hardware and software distinct from the client data processors that permanently/semi-permanently stores, and operates to provide access by the client data processors to, bits, bytes, blocks, bit files, files, and/or file-systems that are not permanently/semi-permanently stored in the client data processor memories. Conceptually, a direct access storage device that is channel-connected to a client data processor can be considered to be a mass storage system. However, the term mass storage system usually applies to a larger and more complicated set of storage equipment and software than is encompassed by a single direct access storage device.

Network file service refers to a method of accessing a file-system by a client data processor via a network when the subject file-system is remotely located from the client data processor. The file-system being accessed is located on a different system that is remote from the accessing client data processor, and a network that interconnects the client data processor with the system that has the subject file-

system is used to communicate requests and data between the two systems. Typically, software is provided on both systems that makes the remote file-system look to users/applications on the client data processor as if it was resident on a direct access storage device which was directly channel-attached to the client data file processor. Generally, the term network file service implies that the remote system that has the subject file-system must execute with its own file-system code, file-system commands that are received via the network from the client data processor's file-system code. Network file service is a different type of service than if device driver commands originating from the client data processor were to be executed on the remote system.

An operating system is software that exists between the user/application and the hardware of the data processor. The basic purpose of an operating system is to allow a plurality of users/applications to simultaneously believe that each has exclusive use of the hardware and operating system of a data processor, and at the same time to present to the users/applications a computing/programming model that is much richer and easier to use than the computing model presented by the raw hardware.

The path name for a file is the sequence of the file names of the directories, starting at the top-most directory in the hierarchy (which is referred to as the root directory), which one must pass through to reach the file in question, including the name of the file-system containing the hierarchy that contains the file, and ending with the file name of the file itself. Because there is only one such path to any particular file, this path name is unique within the hierarchy.

A sequential access storage device is a hardware device which stores a collection of blocks sequentially, and a mechanism which allows the accessing and/or altering of any one of the blocks by first accessing all blocks preceding the subject block. The time and effort to access the nth block on such a device is approximately n times the time and effort to access the first block, once the first block is accessed. Blocks may be added past the last sequential block, or blocks at th end of the sequence may be removed. Blocks on a particular sequential access storage device may vary in size. Blocks on a sequential access storage device are frequently used to archive data because the storage media is relatively

inexpensive. However, sequential devices are poorly suited for the random access needed to access most file-systems. Therefore, the information on sequential devices is usually copied onto a direct access storage device prior to access. Because sequential storage devices are so frequently used as archival devices, the term archival device is used herein to mean a sequential access storage device.

A switching-channel fabric, or more simply a switching-channel or even channel switch, interconnects multiple data processors and device controllers, each of which has a fixed hardware path to the switch. At any given time the switch establishes temporary connections between pairs of connected devices. Multiple pairs may be temporarily connected, but at any given time a given pair is connected, no other device on the switch may connect to that pair.

A user is a human being who interacts with a client data processor and its operating system to perform some desired computing activity using zero or more applications.

Existing Tape Data Storage System Architectures

Figure 1 illustrates in block diagram form the overall architecture of a traditional direct attached tape device configuration, wherein a plurality of data processors DP1-DP3 are each directly connected to at least one tape device TD1-TD3, which tape devices TD1-TD3 for the plurality of data processors DP1-DP3 are served by an automated tape cartridge library system ACS and its associated automated tape cartridge library system controller ACSC. The data processors DP1-DP3 communicate with the automated tape cartridge library system controller ACSC via a bus CN. The automated tape cartridge library system controller ACSC receives data file access requests from the data processors DP1-DP3 and allocates a designated tape cartridge located in the automated tape cartridge library system ACS to serve the request. The automated tape cartridge library system controller ACSC transmits commands to the automated tape cartridge library system ACS to retrieve the selected tape cartridge from its cartridge storage location in the automated tape cartridge library system ACS and mount the retrieved tape cartridge on the tape device TD1-TD3 directly connected to the requesting data processor DP1-DP3. The automated tape cartridge library system controller ACSC transmits commands to the requesting data processors DP1-DP3 when the

selected tape cartridge is mounted in the identified tape device TD1-TD3 to indicate that the requested data is presently available on the identified tape device TD1-TD3.

Figure 2 illustrates in block diagram form the overall architecture of a traditional client/server network attached tape configuration which is similar in operation to the system of Figure 1. The differences are that the data processors DP1-DP3 are no longer directly connected to the tape devices TD1-TD3, but share the plurality of tape devices TD1-TD3 among the data processors DP1-DP3. The communications among the data processors DP1-DP3, tape devices TD1-TD3 and automated tape cartridge library system controller ACSC are effected via a common network N which transmits both data and control information therebetween. Each of the tape devices TD1-TD3 are all located behind a corresponding server TS1-TS3, using a SCSI interface, and is a static configuration, difficult to scale to changing data storage needs.

Figure 3 illustrates in block diagram form the overall architecture of a traditional network attached RAIT tape array configuration where the individual tape devices of Figure 2 are replaced by an array of tape devices TD1-TD4. The tape devices TD1-TD4 are configured in a redundant array of data storage devices in a manner analogous to the Redundant Array of Inexpensive Disks (RAID) which is a well known architecture in the disk device technology. The operation of a RAID system, and by extension, the corresponding operation of the RAIT system is not described in detail herein, since such information is readily available in the literature. The RAIT system includes a RAIT controller RC which is connected to the network N via a plurality of servers RS1-RS2. The traditional RAIT is located behind a server, using a SCSI interface, and is a static configuration, difficult to scale to changing data storage needs.

Figure 4 illustrates in block diagram form the overall architecture of a prior art network connected mass storage system which uses a multitude of data storage devices, which is described in additional detail below.

Mass Storage Systems

The prior art mass storage system 10 described in the Sloan et al paper is implemented using three different communication paths as shown in Figure 5. The

bit file server 15 is the sole interface to the mass storage system 10 by client data processors 1-n, and manages all bit file aspects of client data processors' file-systems, such as file-system mounting and dismounting. On the other hand, all access by client data processors of file-system data blocks occurs through the channel switching-fabric 11, without intervention by the bit file server 15. In particular, the main task of the bit file server 15 is to manager the task of staging, that is copying file-systems back and forth between archival storage devices and direct access storage devices, and maintaining logical connections between client data processors 1-n and file-systems resident on direct access storage devices.

A network mount request/relay path 50 connects operating systems resident on client data processors 1-n to the bit file server 15, and is the sole path between mass storage system 10 and the client data processors 1-n for passing information related to the staging, that is copying, of file-systems between archival storage devices and direct access storage devices, and for passing information related to the management of logical connections between the client data processors 1-n and file-systems resident on mass storage system 10 direct access storage devices. The network mount request/reply path 50 is implemented as a FDDI ring on mass storage system 10, though other network technologies could be used as well.

Under the command of the bit file server 15, the storage servers 60-1 to 60-m manage individual archival and direct access storage devices. Each storage server is responsible for maintaining mapping information regarding the collection of file-systems on its associated storage device. In the case of a direct access storage device, the associated storage server must maintain a map of which disk blocks are currently allocated to which file-systems, and which blocks are available for new use. In the case of an archival device, the associated storage server must maintain a map of which media portions contain which file-systems, and which media portions are available for new use. Under command of the bit file server 15, storage servers also direct the actions of their associated storage devices regarding file-system staging (copying). This direction occurs through special paths 18-1 and 18-m that exist between each storage server and its associated storage device. Depending upon the intelligence of the storage device's controller, direction by a storage server of a copy operation may be very minimal or it may be quite

extensive. In some cases, a single command to the storage device from its storage server may be sufficient, while in other cases, the storage server must issue a command for each data block to be transferred. Additionally in some cases a pair of storage servers may need to communicate extensively with each other as well as their own associated devices.

A control path 13 is provided to connect the bit file server 15 and the data storage servers 60-1 to 60-m to one another. This control path 13 is not physically or logically accessible by the client operating systems 1A resident on client data processors 1-n. Because a high communications bandwidth is not needed for control path activity, the control path 13 can be implemented by means of an Ethernet network which is compatible with many commercially available data storage subsystems, although other network technologies could be used as well.

A third communication path is a high speed channel switching fabric 11 that provides switched-channel connections among data storage devices 40-1 to 40-m and the client data processors 1-n. The channel switching-fabric 11 is implemented using a high speed switch featuring HIPPI-compatible channels, though other network technologies could be used as well.

Importance of the Channel-Switching Fabric for File-System Access

There is an important distinction between a network and a channel. A network is a general purpose data communications path which connects a plurality of client data processors to one another for general communications purposes. Data that moves through such a network is usually processed through many layers of software. Since both the network hardware and the network software is general purpose, neither is optimized for any particular task. In particular, general purpose network hardware and software is generally incapable of connecting client data processors and remote data storage devices at data communication transfer rates capable of driving such data storage devices at their full data transfer rates.

A channel is a special purpose communications path which is designed to connect a client data processor to a data storage device at very high data transfer rates. Data that moves through a channel is handled with simple, special-purpose, lower level of protocol and therefore with lower overhead than with a network.

The mass storage system 10 enables users and client applications to access file-systems through what appears to be a directly attached storage device via a conventional block-access storage device driver 300, yet no performance penalty occurs as would be the case with any method of accessing a storage device across a network. It is transparent to the users, applications, operating systems, client data processors, and the storage and archival devices that a plurality of block I/O requests and data for a plurality of file-systems and a plurality of staging operations move through a high performance channel switching-fabric 11, which is simultaneously handling the switching of channel connections between a plurality of client data processors 1-n and a plurality of data storage devices 40-1 to 40-m.

To effect such transparency as provided by mass storage system 10, the only major modifications necessary to the client data processor 1 operating system 1A are modifications to the client data processor device driver 300 attached to the channel switching-fabric 11. Generally, device driver modifications are not difficult since most operating systems are designed to have new device drivers easily added. The mounting code 17 is not a part of the operating system; it is generally installed as a user-level process. Therefore, the operating system code is not affected by the addition of the mounting code 17, although it is expected that slight modifications to the file-system code 203 may be necessary. Even here, the modifications are not expected to be too difficult because most file-system codes already include provisions for distinguishing between locally and remotely mounted file-systems, and also include provisions for communicating with mounting codes somewhat similar to mounting code 17.

A difficulty with the mass storage system architecture is that it is designed to interconnect archive data storage systems with the local connected data storage devices which server the data processors. There is no analogous use of the high bandwidth connection in the realm of the local connected data storage devices. Thus, the data processors are served by the traditional tape device configurations noted above, since the high bandwidth network interconnection of the mass storage systems is not implemented on a local level. In addition, there is no use of the virtual device concept in the mass storage system, where the present data needs of the data processors are used to customize the data storage operation.

System Architecture of Distributed Virtual Tape Storage Subsystem

By pooling the tape devices TD1-TD5 together, and interconnecting them with the data processors DP1-DP4 via a fiber data network FN, these problems are overcome. This architecture uses shared pooling of tape device resources to realize multiple virtual devices, which are available to any of the data processors DP1-DP4. This architecture also enables any tape array to be realized using the tape devices TD1-TD5, including, but not limited to: RAIT 0, 1, 3, 4, 5. The tape array can be reconfigured dynamically since the network FN switchably interconnects the tape devices TD1-TD5 to the data processors DP1-DP4 under control of a networked storage manager NSM. The bandwidth of this networked distributed virtual tape device data storage subsystem is scalable and can be changed on demand and network security can be implemented.

The use of a networked storage manager NSM enables the storage devices to be managed as a pool and yet the devices are directly attached to the network FN as individual resources. The networked storage manager NSM must provide the mechanism for the enterprise management to control storage device allocation and configuration as well as other functions, such as cartridge movement and data migration. The rules which could be implemented address response time constraints, data file transfer size, data file transfer rates, data file size bounds and the like. The networked storage manager NSM manages the allocation, configuration and security.

The networked storage manager NSM for the distributed virtual tape data storage subsystem contains a plurality of software elements including: resource allocation RA, resource configuration RC, and resource management RM. It may also contain security software for authentication. The resource allocation RA software has the responsibility to keep track of the resource usage such as which data processor DP1 presently owns a tape device TD3, which of the plurality of tape devices TD1-TD5 are free to be allocated to requesting data processors and the like. The resource configuration RC software allows the operator to configure the resources for the data processors DP1-DP4 which are attach d to the network FN. The operator can assign the maximum number of tap devices that a data processor can have and also define the configuration of the tape devices. The

resource configuration RC software automatically configures both the tape devices TD1-TD5 allocated to a data processor DP1-DP4 as well as the connection between the data processor and a tape device(s). The resource management RM software queues the request for the resource allocation and notifies the data processor when the requested resource is ready, or it can schedule the availability of the resources.

Fiber Channel Network

A SCSI interface is a bus topology, and all devices that are connected to the SCSI bus share the bus bandwidth, with only a single device being capable of accessing the bus at a time. As more devices are attached to the SCSI bus, the contention for the bus increases and this technology prevents the creation of a scalable system. The SCSI interface is not designed for switched point-to-point operation, although it can be converted to this use by the implementation of a SCSI-to-fiber bridge. In contrast to the SCSI bus, the fiber channel fabric is capable of data rates (1 Gbit/sec) far in excess of the SCSI bus and is also expandable to up to 190 ports.

In order to incorporate the SCSI bus into this distributed virtual tape storage subsystem architecture, each SCSI device (DP1-DP4, TD1-TD5) is equipped with a SCSI-to-fiber channel bridge CB1-CB9, and all of the SCSI-to-fiber channel bridges CB1-CB9 are connected to the fiber channel network FN. Each SCSI-to-fiber channel bridge CB1-CB9 has its own unique Bridge Node ID and each device connected to a SCSI-to-fiber channel bridge CB1-CB9 also has its own SCSI ID. The nexus for a logical connection is of the form:

<HOST SCSI ID><HOST BRIDGE ID><TARGET BRIDGE ID><TARGET SCSI ID>

The fiber channel network FN is controlled by means of a network resource allocation RA manager which is embedded in the automated cartridge library system controller ACSC. The network resource allocation RA manager software executes on a separate device, such as a SPARC work station, which is connected to the fiber channel network FN via a SCSI-to-fiber bridge CB0. The automated cartridge library system controller ACSC and all of the data processors DP1-DP4 are interconnected via an Ethernet control network CN.

In operation, a data processor DP1 initiates a request to access a selected data volume which is managed by the automated tape cartridge library system ACS. The data processor DP1 transmits this request to the automated tape cartridge library system software which resides in the automated cartridge library system controller ACSC via the control network CN. The automated tape cartridge library system software receives the request and ascertains the physical location of the physical medium which contains the selected data volume. The automated tape cartridge library system software also identifies an available tape device TD4 which is served by the automated tape cartridge library system ACS and which is connected to the fiber channel network FN. The automated tape cartridge library system software transmits control signals to the automated tape cartridge library system ACS to initiate the mounting of the physical medium which contains the selected data volume on the identified tape device TD4. Once the tape mount is successful, the automated tape cartridge library system software notifies the network resource allocation RA manager software via the control network CN and the network resource allocation RA manager software transmits a RESERVED command to the SCSI-to-fiber channel bridge CB8 which serves the identified tape device TD4, to reserve the identified tape device TD4 for the data processor DP1 that has the network address <HOST SCSI ID><HOST BRIDGE ID>. The network resource allocation RA manager software also transmits a TARGET BRIDGE ID command to the SCSI-to-fiber channel bridge CB1 that serves the requesting data processor DP1 to thereby identify the SCSI-to-fiber channel bridge CB8 that serves the identified tape device TD4. Once the network resource allocation RA manager software reserves the identified tape device TD4 for the requesting data processor DP1, it returns a status message to the automated tape cartridge library system software indicating a successful status. The requesting data processor DP1 can now transmit SCSI commands to the identified tape device TD4 via the fiber channel network FN. Data processor DP1 has device driver software that satisfies the various virtual device configurations of the tape devices TD1-TD5 that are part of the virtual tape data storage subsystem.

When the requesting data processor DP1 completes the desired operations, the requesting data processor DP1 transmits a DISMOUNT command via the

control network CN to the automated tape cartridge library system software. Upon receipt of the DISMOUNT command, the automated tape cartridge library system software notifies the network resource allocation RA manager software, also via the control network CN, of the received DISMOUNT command. The network resource allocation RA manager software transmits a RELEASE command to the SCSI-to-fiber channel bridge CB9 that serves the identified tape device TD4. The automated tape cartridge library system software deallocates the identified tape device TD4 in its resource table and returns a successful status to the requesting data processor DP1 via the control network CN. The identified tape device TD4 is then available to be used by any data processor.

Tape Array Configuration

If the request issued by the data processor DP1 is for a tape array configuration of the data storage subsystem, such as RAIT 3 (3 data drives +1 parity drive), the number of tape devices allocated is 4. One tape device is the parity storage device while the remaining tape devices are the data storage devices. Under normal tape array operation, the data processor DP1 transmits the data and the data processor generated parity information to the tape array. The data processor DP1 manages the distribution of the data, for example, block striping to distribute the data across the four tape devices which comprise the tape array. In this instance, block 1 of the data is sent by data processor DP1 to data device 1, block 2 of the data is sent by data processor DP1 to data device 2, block 3 of the data is sent by data processor DP1 to data device 3 and the parity block is generated by the data processor DP1 XORing data block 1, data block 2, data block 3. The parity block is then transmitted by data processor DP1 to the parity data device. In this process, the data processor DP1 transmits four blocks of data to the tape array.

A more efficient process uses the capability of the fiber channel network FN to multicast the data and the tape device ability to XOR the received data blocks. At the time that the data block 1 is transmitted by data processor DP1 to a first tape device TD1, this data block is also transmitted by data processor DP1 to the parity tape device (ex- TD5), by instructing the fiber channel network FN to multicast the data block to the parity tape device TD5. Similar operations are performed with

data block 2 and data block 3 and their corresponding data storage tape devices. Once the last data block is received by the parity tape device TD5, these data blocks are XORed by the parity tape device TD5 to produce the parity block of data. The parity block of data is then written by the parity tape device TD5 to the recording medium. The number of data blocks transmitted by data processor DP1 through the fiber channel network FN in this process are three rather than the four of the previous process. Data processing bandwidth is therefore conserved.

Tape Array Features

Figures 6 and 7 illustrate in block diagram form the overall architecture of the dynamic mapping and data striping functions, respectively, of the network attached virtual tape storage subsystem. In particular, Figure 6 illustrates the concept of interconnecting the data processors DP1, DP2 with a plurality of data storage subsystem images. The plurality of tape devices TD1-TD5 can be configured in any of a number of subsystem configurations. Figure 6 illustrates that one TD1 of the tape devices TD1-TD5 can be directly connected to the data processor DP1 or a plurality TD3-TD5 of the tape devices TD1-TD5 can be connected to the data processor DP1 in a RAIT 3 configuration. The configuration of the tape devices TD1-TD5 is effected by the networked storage manager NSM on a dynamic basis as a function of the data storage needs of the data processor DP1. Similarly, data processor DP2 can be interconnected with tape devices TD2, TD4 in a RAIT 1 configuration as well as to tape devices TD2, TD4, TD1 in a RAIT 5 configuration. The particular tape devices which are selected for the interconnection are selected dynamically by the networked storage manager NSM.

Figure 7 illustrates the concept of striping the data across a plurality of the tape devices TD1-TD5. In the example of Figure 7, the data is written concurrently to two tape devices at a time, using the multicast capability of the fiber channel network FN. The selection of the two tape devices is controlled by the networked storage manager NSM by simple designating different fiber channel network ports for the writing of the data. Thus, the striping of the data can be to tape devices TD2, TD3 via ports 4, 5 and thence to tape devices TD4, TD5 via network ports 8, 2.

WHAT IS CLAIMED:

1. A distributed virtual tape data storage system for storing data files on a plurality of tape data storage elements for access by at least one data processor, comprising:

a plurality of tape drive means for providing said at least one data processor with access to said data files stored on said plurality of tape data storage elements;

tape library means for robotically storing and retrieving said plurality of tape data storage elements for mounting on said plurality of tape drive means;

network means, interconnecting said at least one data processor and said plurality of tape drive means, for exchanging data therebetween; means, connected to said tape library means, for managing storage of data on said plurality of tape data storage elements mounted in said plurality of tape drive means by said tape library means;

means connected to said at least one data processor and said means for managing storage of data, for transporting control signals therebetween;

wherein said means for managing storage of data comprises:

means for storing mapping data indicative of a correspondence between data processor data files and physical data storage locations in said plurality of tape data storage elements used to store said data files;

means, responsive to a one of said data processors requesting access to an identified data file, for selecting ones of said tape data storage elements which contain said identified data file based upon said data file mapping data; and

means for activating said tape library means to mount said selected tape data storage elements on selected ones of said plurality of tape drive means for access of said identified data file by said requesting data processor via said network means.

2. The distributed virtual tape data storage system of claim 1 wherein said means for managing storage of data further comprises:

means, responsive to a one of said at least one data processor requesting storage of a selected data file on said plurality of tape data storage elements, for allocating a data storage subsystem image for the storage of said selected data file.

3. The distributed virtual tape data storage system of claim 2 wherein said means for managing storage of data further comprises:

means for activating a plurality of said tape drive means to implement said data storage subsystem image for the storage of said selected data file.

4. The distributed virtual tape data storage system of claim 3 wherein said means for managing storage of data further comprises:

means for transmitting data to said at least one data processor to identify said activated plurality of said tape drive means to receive said selected data file transmitted by said at least one data processor via said network means.

5. The distributed virtual tape data storage system of claim 4 wherein said means for transmitting data transmits an identification of a port on said network means for each of said activated plurality of said tape drive means.

6. The distributed virtual tape data storage system of claim 1 further comprising:

a plurality of network interface means, each connected between a one of said plurality of tape drive means and said network means for interfacing said tape drive means with said network means.

7. The distributed virtual tape data storage system of claim 6 wherein said tape drive means includes a SCSI interface and said network means comprises a fiber channel network, said network interface means comprise a SCSI to fiber channel converter.

8. The distributed virtual tape data storage system of claim 1 wherein said network means comprises a multicast network for concurrently transmitting data to a plurality of said tape drive means connected to said network means.

9. The distributed virtual tape data storage system of claim 1 wherein said network means comprises a nonblocking high bandwidth switching network.

10. A method for managing a distributed virtual tape data storage system for storing data files on a plurality of tape data storage elements for access by at least one data processor, said distributed virtual tape data storage system comprising a plurality of tape drives for providing said at least one data processor with access to said data files stored on said plurality of tape data storage elements, a tape library for robotically storing and retrieving said plurality of tape data storage elements for mounting on said plurality of tape drives, a network, interconnecting said at least one data processor and said plurality of tape drives, for exchanging data therebetween, wherein said method comprises the steps of:

storing mapping data indicative of a correspondence between data processor data files and physical data storage locations in said plurality of tape data storage elements used to store said data files;

selecting, in response to a one of said data processors requesting access to an identified data file, ones of said tape data storage elements which contain said identified data file based upon said data file mapping data; and

activating said tape library to mount said selected tape data storage elements on selected ones of said plurality of tape drives for access of said identified data file by said requesting data processor via said network.

11. The method of claim 10 wherein said method further comprises the step of:

allocating, in response to a one of said at least one data processor requesting storage of a selected data file on said plurality of tape data storage elements, a data storage subsystem image for the storage of said selected data file.

12. The method of claim 11 wherein said method further comprises the step of:

activating a plurality of said tape drives to implement said data storage subsystem image for the storage of said selected data file.

13. The method of claim 3 wherein said method further comprises the step of:

transmitting data to said at least one data processor to identify said activated plurality of said tape drives to receive said selected data file transmitted by said at least one data processor via said network.

14. The method of claim 13 wherein said step of transmitting data comprises:

transmitting an identification of a port on said network for each of said activated plurality of said tape drives.

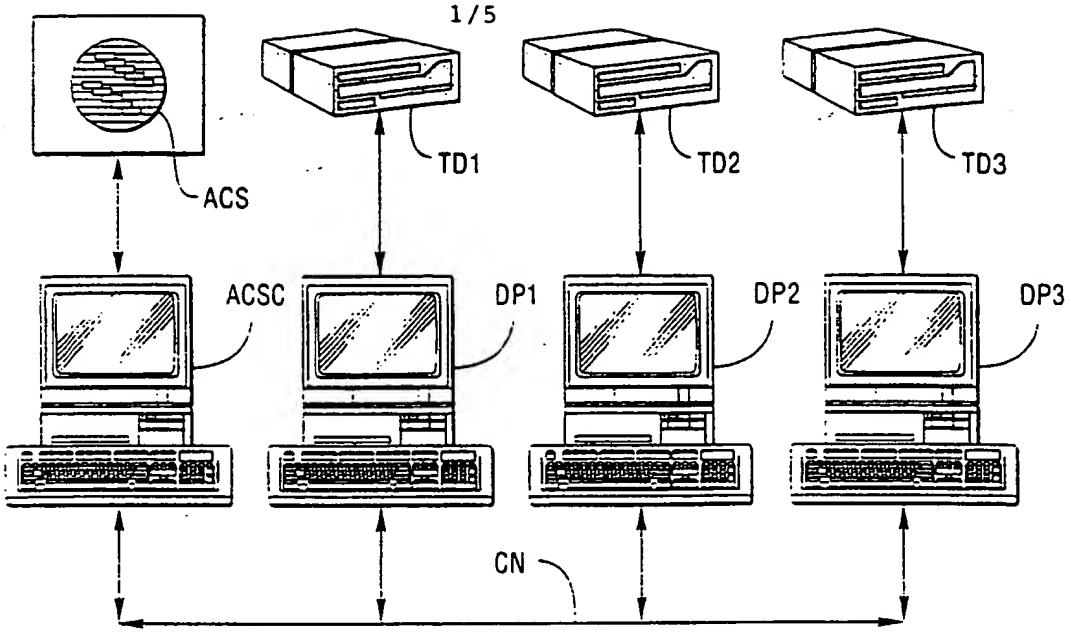

ACS
TD1
TD2
TD3
ACSC
DP1
DP2
DP3
CN

Fig. 1

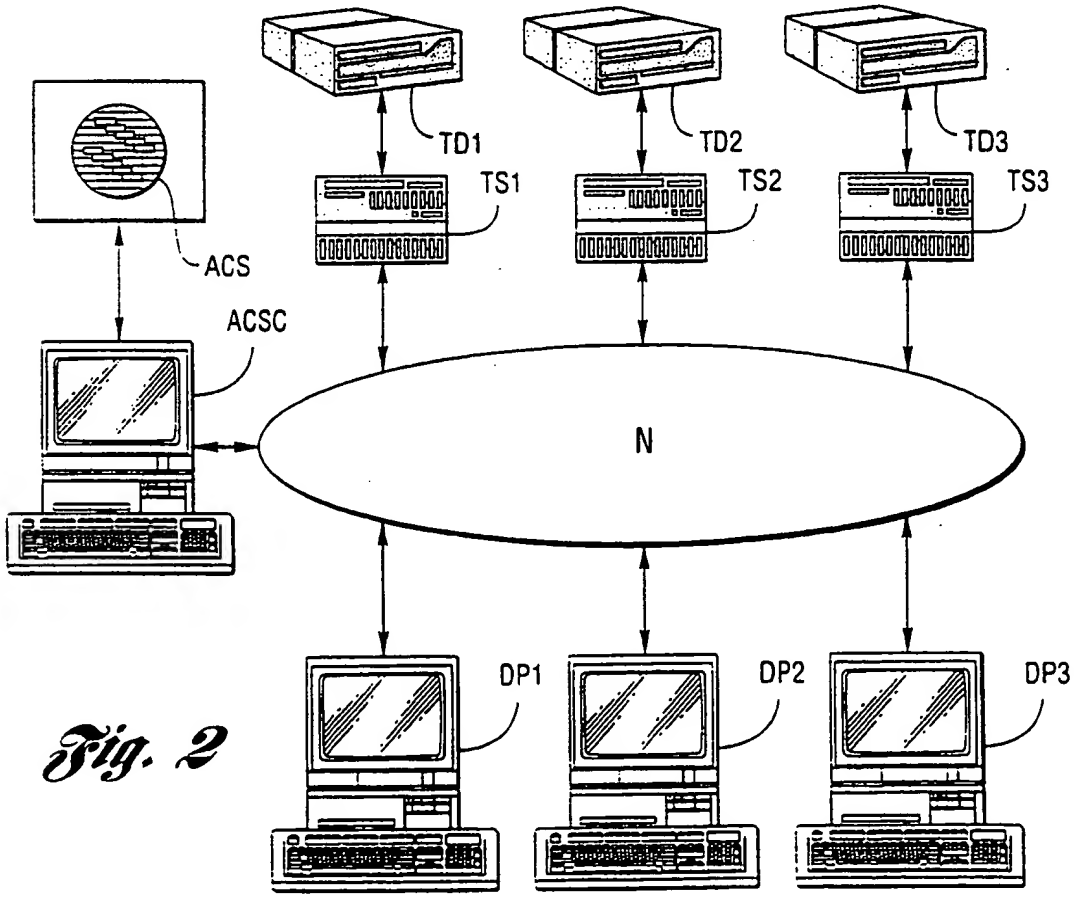

TD1
TD2
TD3
TS1
TS2
TS3
ACS
ACSC
N
DP1
DP2
DP3

Fig. 2

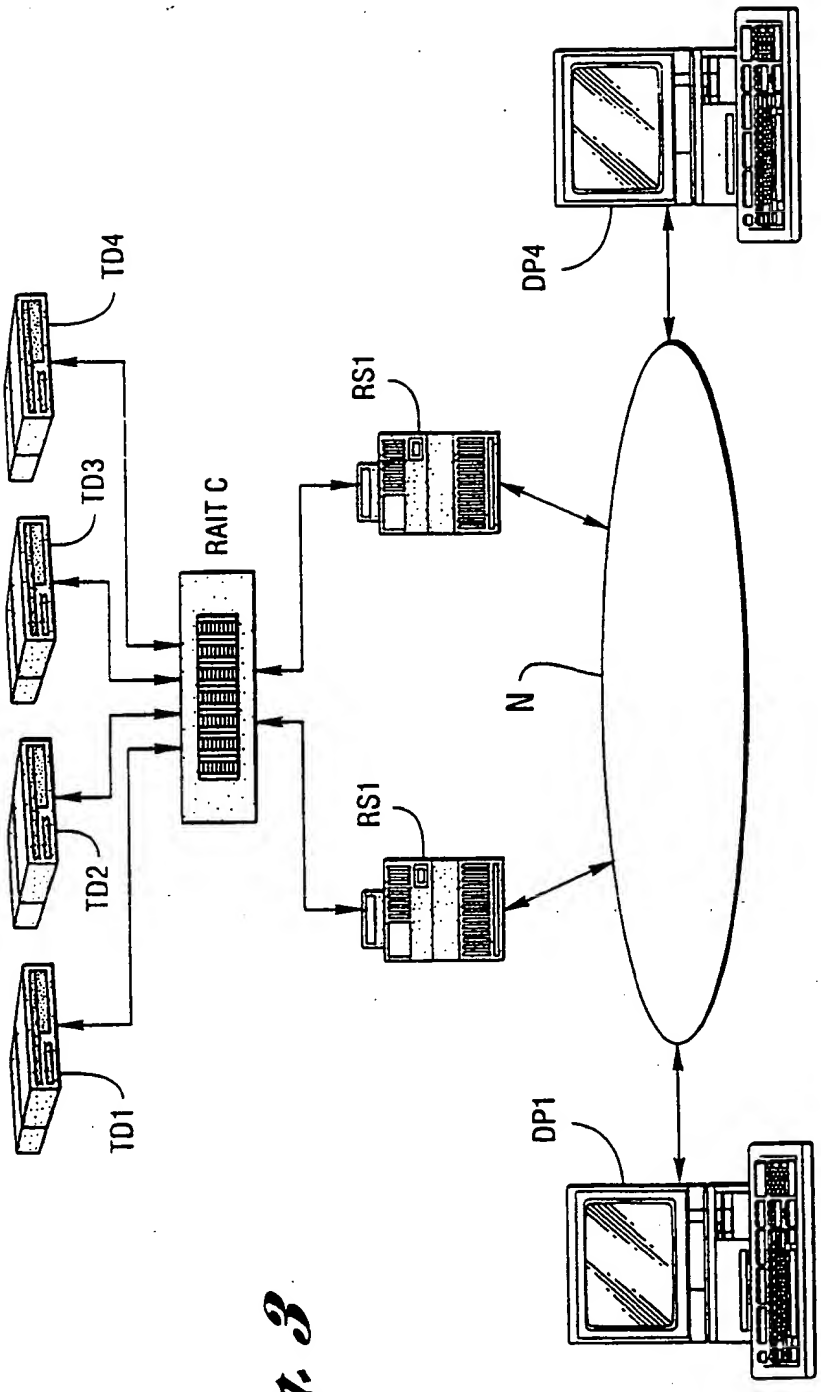


Fig. 3

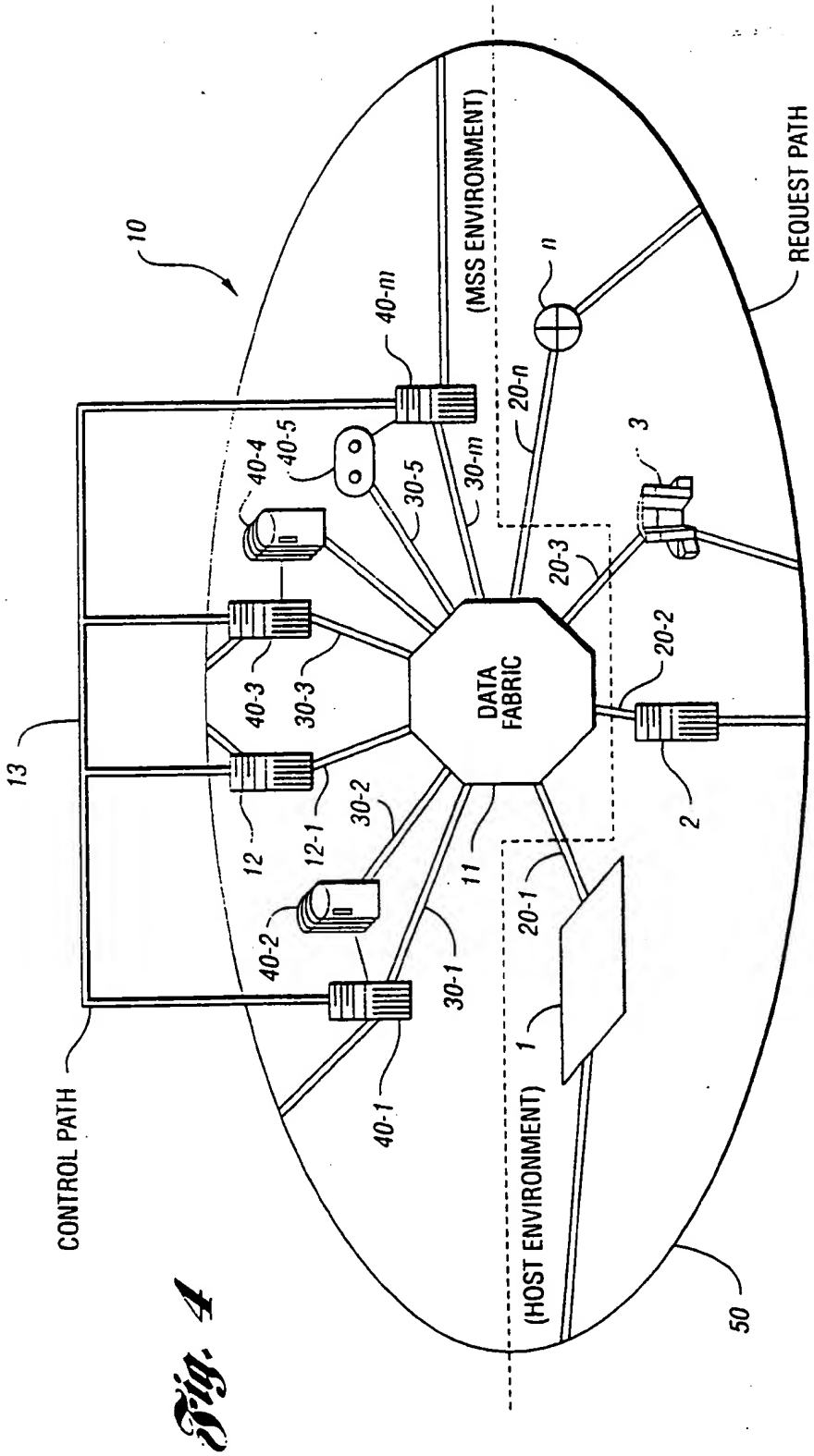

10
(MSS ENVIRONMENT)
REQUEST PATH
40-m
n
20-n
30-m
40-5
40-4
30-5
3
20-3
DATA FABRIC
20-2
40-3
30-3
13
12
12-1
30-2
11
20-1
2
40-2
30-1
1
40-1
CONTROL PATH
(HOST ENVIRONMENT)
50

Fig. 4

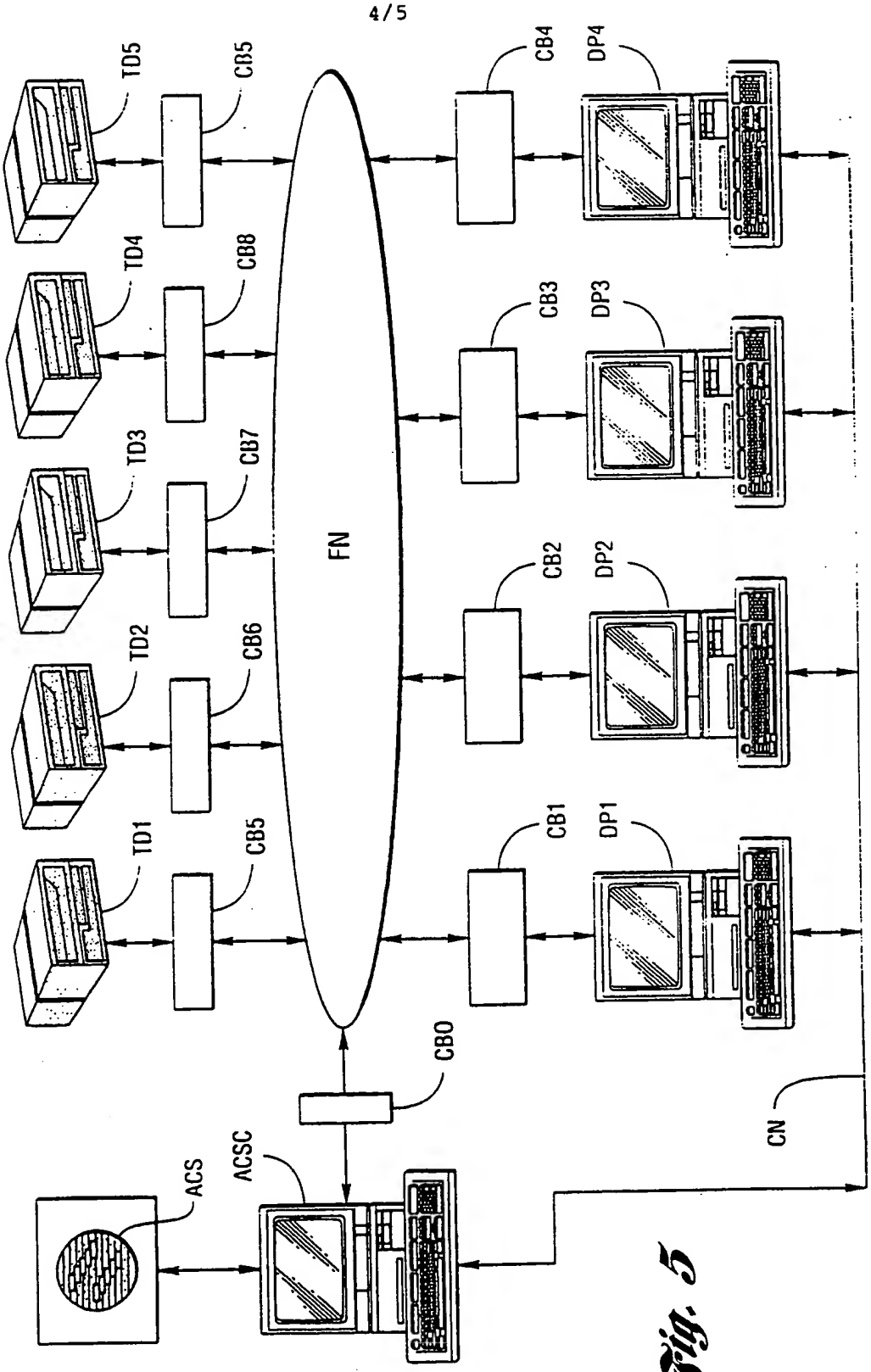

TD5
CB5
CB4
DP4
TD4
CB8
CB3
DP3
TD3
CB7
FN
CB2
DP2
TD2
CB6
TD1
CB5
CB1
DP1
CBO
CN
ACS
ACSC

Fig. 5

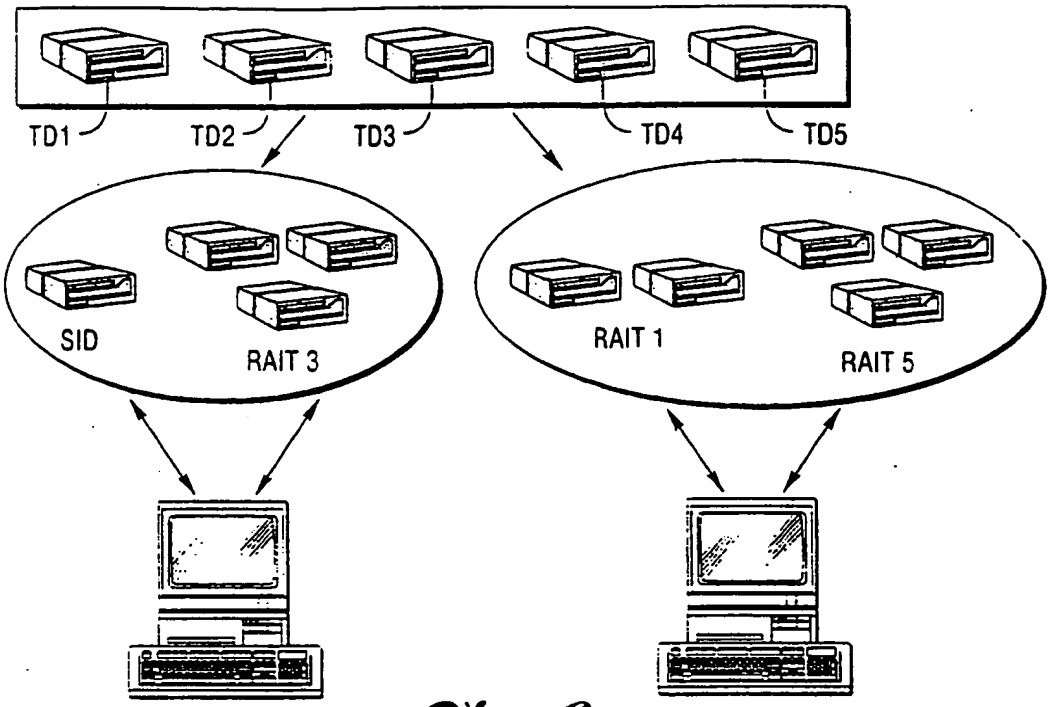

TD1
TD2
TD3
TD4
TD5
SID
RAIT 3
RAIT 1
RAIT 5

Fig. 6

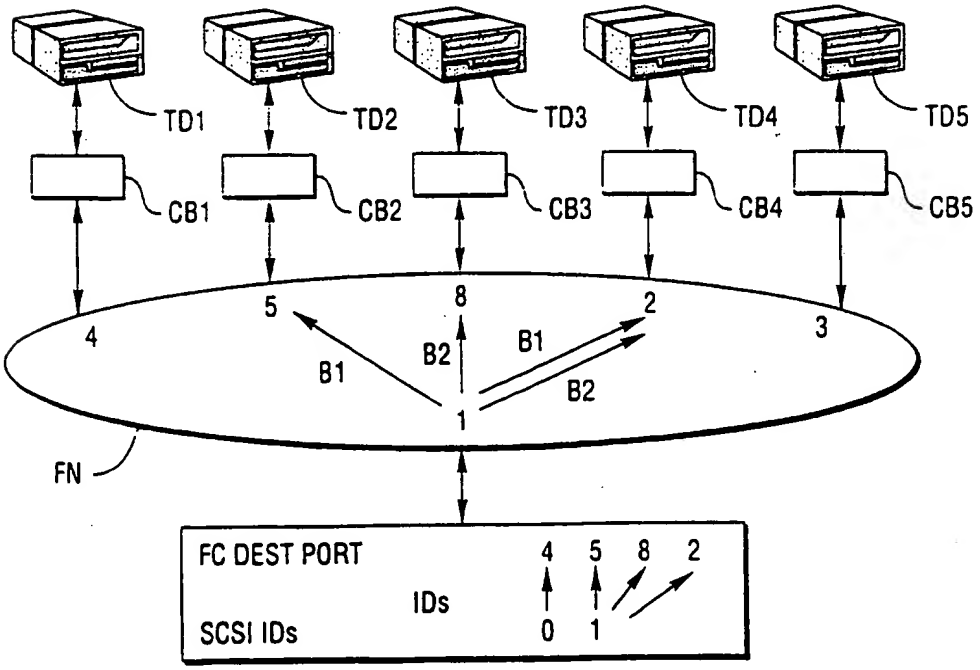

TD1
TD2
TD3
TD4
TD5
CB1
CB2
CB3
CB4
CB5
4
5
8
2
3
B1
B2
B1
B2
1
FN
FC DEST PORT
IDs
SCSI IDs
4 5 8 2
0 1

Fig. 7

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G11B 17/22 | A3

(11) International Publication Number: WO 98/40810

(43) International Publication Date: 17 September 1998 (17.09.98)

(21) International Application Number: PCT/US98/04910

(22) International Filing Date: 12 March 1998 (12.03.98)

(30) Priority Data:
08/815,739 12 March 1997 (12.03.97) US

(71) Applicant: STORAGE TECHNOLOGY CORPORATION [US/US]; 2270 South 88th Street, Louisville, CO 80028 (US).

(72) Inventors: NGUYEN, Thai; 2638 E. 102nd Avenue, Thornton, CO 80229 (US). RAYMOND, Robert, Michael; 6133 Songbird Circle, Boulder, CO 80303 (US).

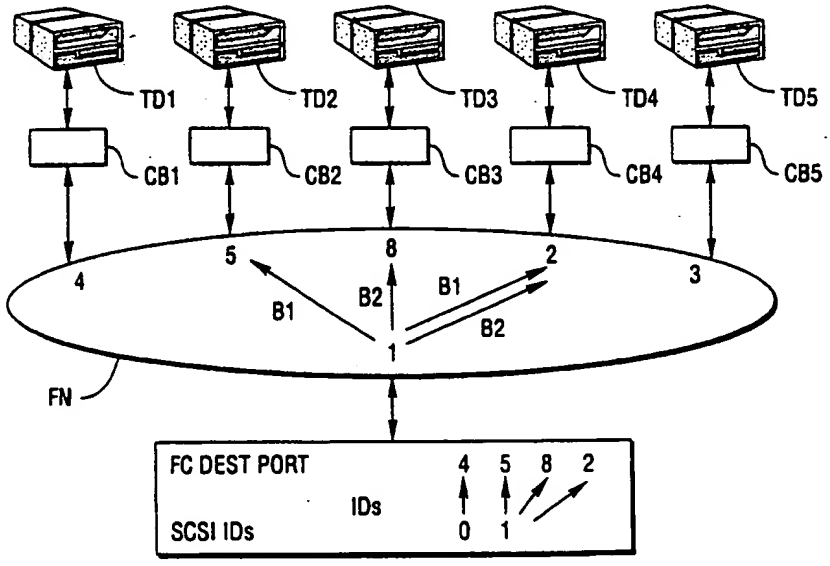
(74) Agents: SCHWARTZ, Paul, M. et al.; Brooks & Kushman, 22nd floor, 1000 Town Center, Southfield, MI 48075 (US).

(81) Designated States: JP, European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(88) Date of publication of the international search report:
29 October 1998 (29.10.98)

(54) Title: NETWORK ATTACHED VIRTUAL TAPE DATA STORAGE SUBSYSTEM



The network attached virtual tape storage subsystem interconnects a plurality of tape devices (TD1-TD5) with a plurality of data processors via a high bandwidth switching network (FN) to implement a virtual, distributed tape data storage subsystem. The virtual, distributed tape data storage system realizes multiple virtual devices, which are available to any of the data processors and the bandwidth of the system is scalable and can be changed on demand. The virtual tape storage subsystem is managed by a system controller which contains a plurality of software elements. The use of a networked storage manager enables the tape devices to be managed as a pool and yet attach the tape devices directly to the network as individual resources. The networked storage manager must provide the mechanism for the enterprise management to control tape device allocation and configuration as well as other functions, such a tape cartridge movement and data migration.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US98/04910

A. CLASSIFICATION OF SUBJECT MATTER
IPC(6) :G11B 17/22
US CL : 369/34, 711/4, 111,114
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
U.S. : 369/34, 711/4, 111,114

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,566,331 A (IRWIN JR. et al.) 15 October 1996, see entire document, especially figures 1&2.	1-14
Y	US 4,945,429 A (MUNRO et al.) 31 July 1990, see entire document, especially cols. 2-4.	1-14
Y,P	US 5,630,067 A (KINDELL et al) 13 May 1997, see Figure 1 and Summary.	1-14
Y	US 5,214,768 A (MARTIN et al.) 25 May 1993, see entire document, especially cols. 5-6.	1-14
Y	US 5,506,986 A (HEALY) 09 April 1996, see col 3, line 49 to col. 5, line 38.	1-5, 10-14

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 19 JUNE 1998	Date of mailing of the international search report 02 SEP 1998
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer P. BATAILLE Telephone No. (703) 305-0134

Form PCT/ISA/210 (second sheet)(July 1992)*